iPhone Views and Animations

How to efficiently make the pretty

Session 132

These are confidential sessions—please refrain from streaming, blogging, or taking pictures.

iPhone Views and Animations

How to efficiently make the pretty

Dan Keen
iPhone Applications Engineer

Demo

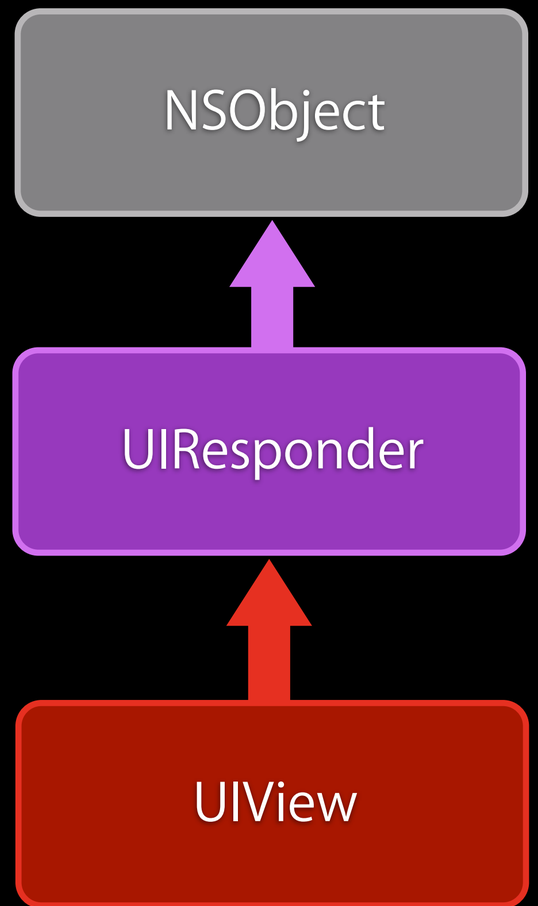
System Emulator

Views

Controls

Animations

UIView



Handles event forwarding
(responder chains)

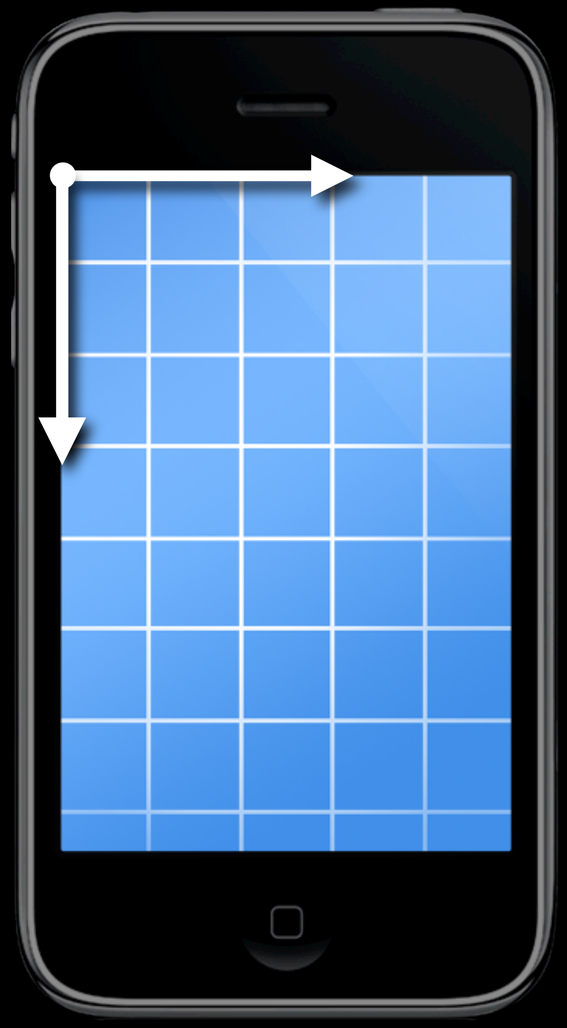
Handles drawing content

Processing Multi-Touch Events on iPhone

Pacific Heights
Thursday 9:00AM

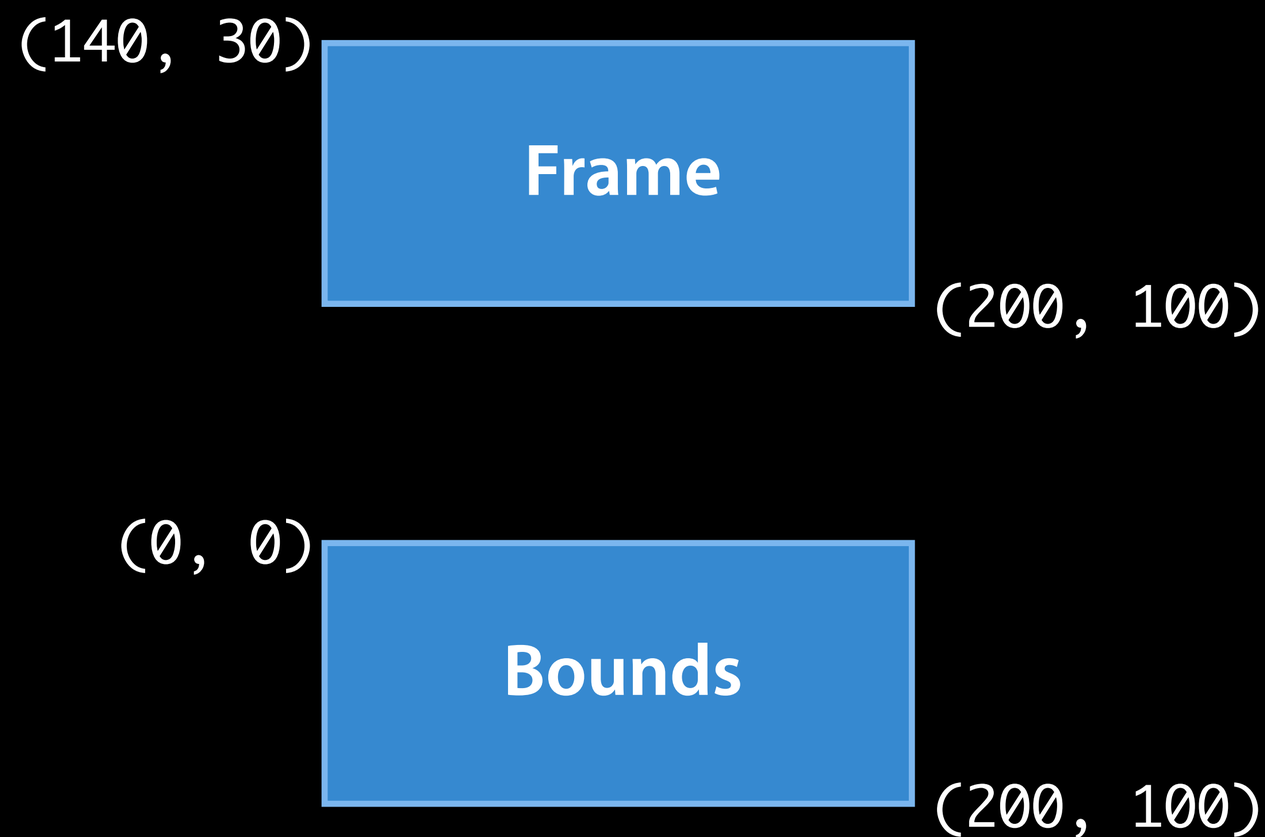
Coordinate System

Origin is at top left



UIView

Frame, bounds, and you



UIView

Coordinates with transformations

	Origin	Size
Bounds	(0, 0)	(200, 100)
Frame	(133.934, -26.066)	(212.132, 212.132)
Center	(240, 80)	

Combining Views

Clipping to bounds

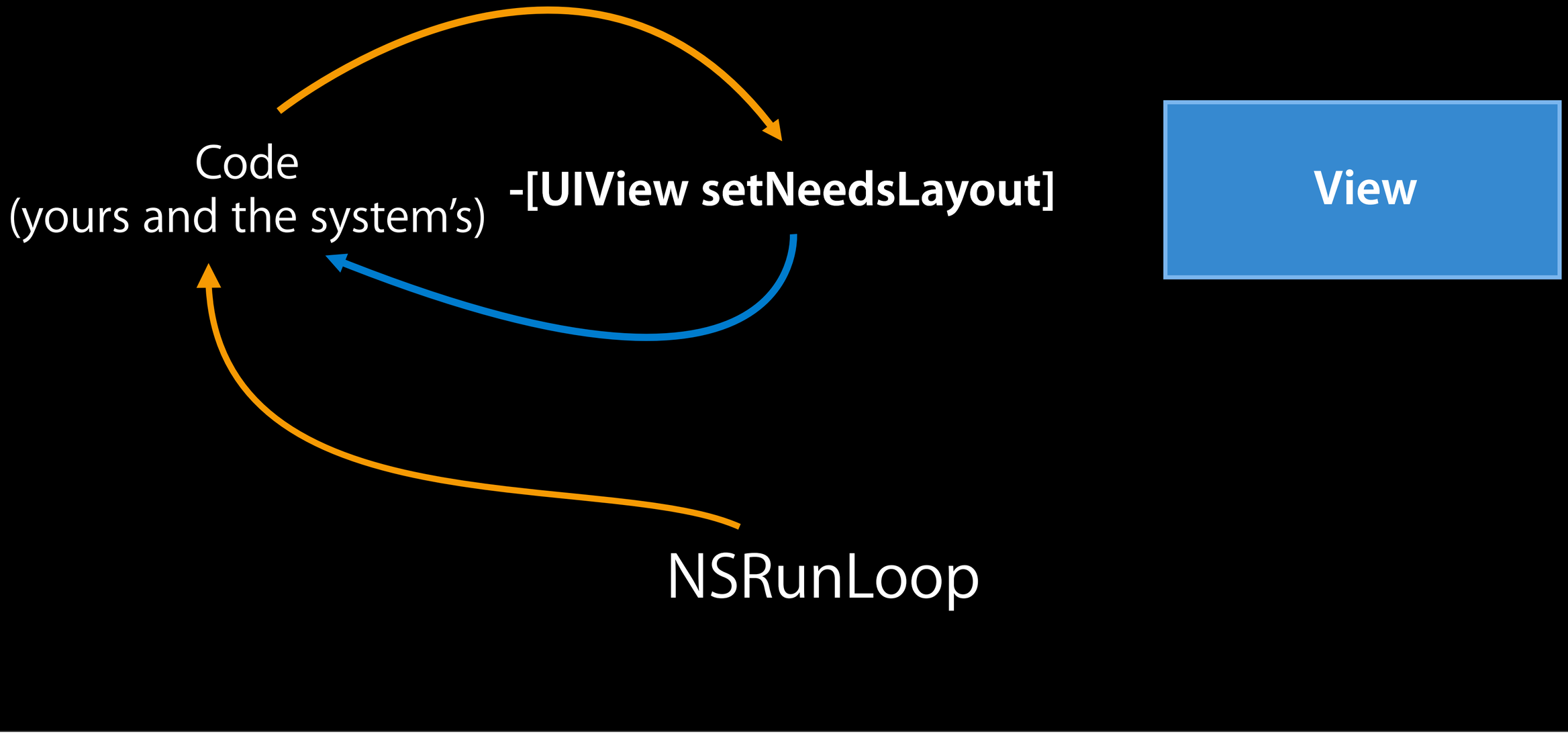
- -[UIView setClipsToBounds:YES]
- Expensive to calculate
- Off by default

Positioning Views

-[UIView layoutSubviews]

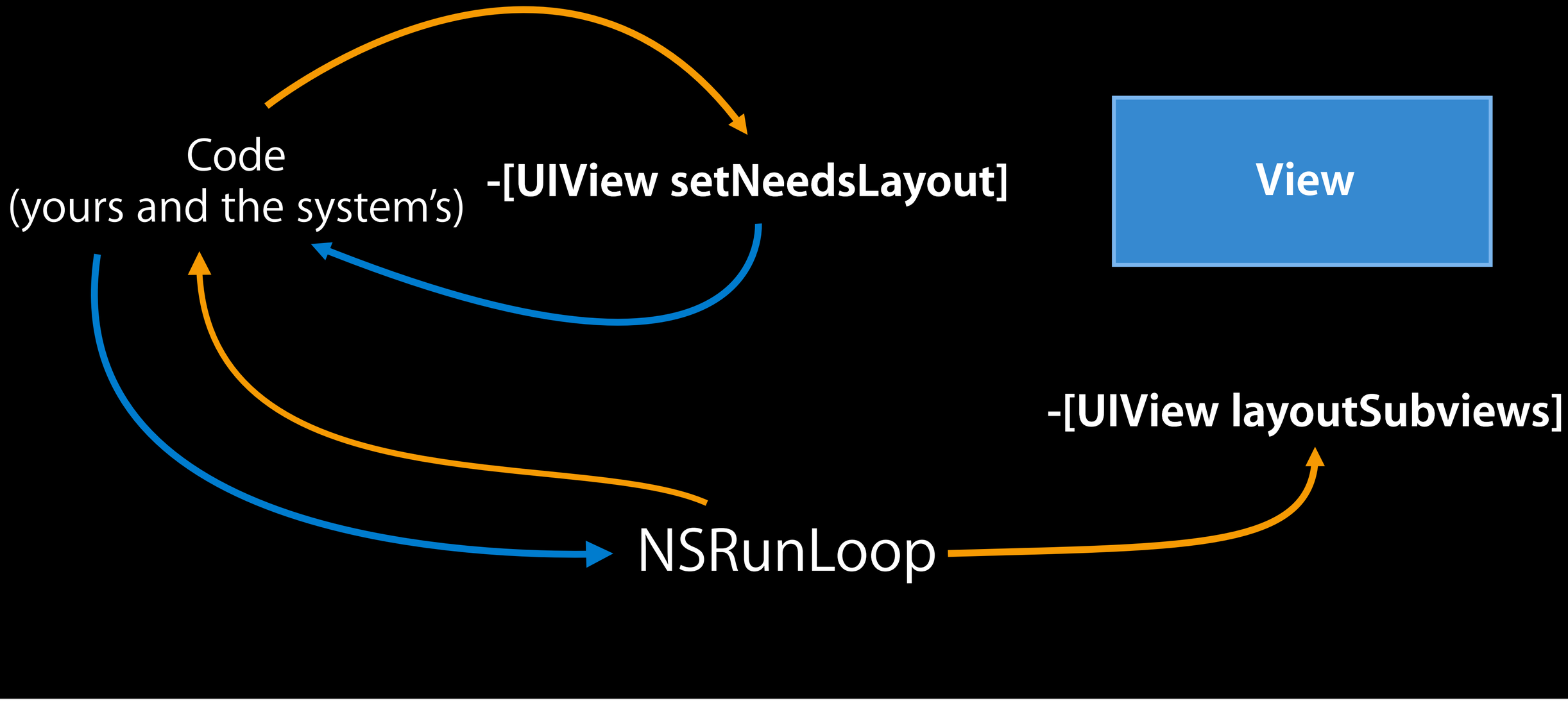
Positioning Views

-[UIView layoutSubviews]



Positioning Views

-[UIView layoutSubviews]



Positioning Views

-[UIView layoutSubviews]

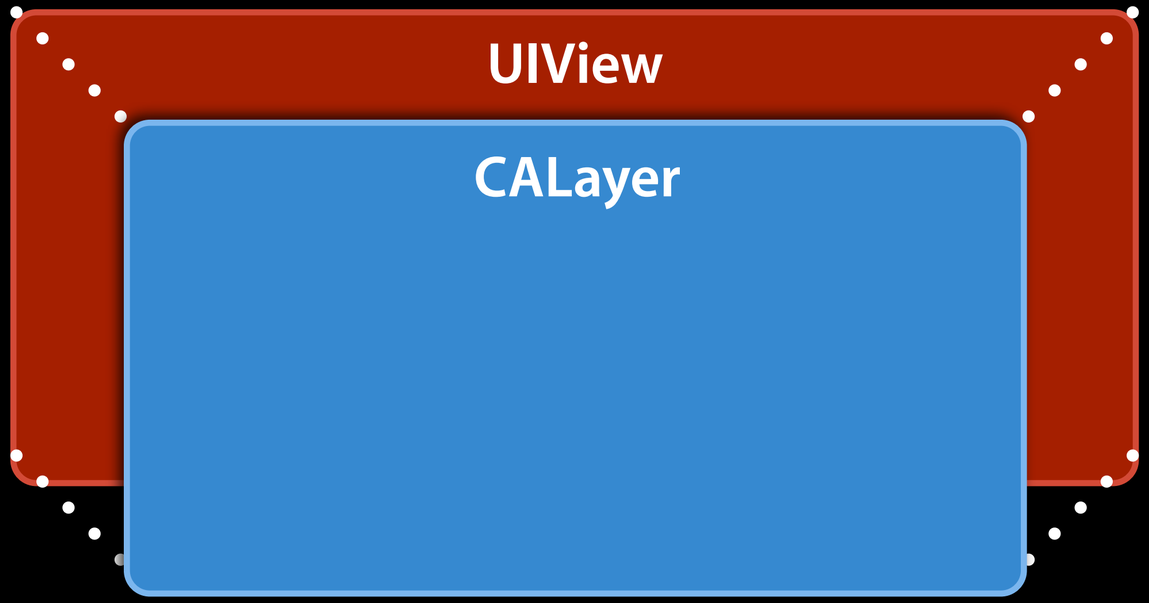
- Don’t position subviews in loadView or initWithFrame:
  - You’ll regret it later…
- Don’t ever call -layoutSubviews directly
  - Except for -[super layoutSubviews], of course!
- If you really, really need a layout now, use -[UIView layoutIfNeeded]
  - Will return early if the view is already laid out properly
- The system will call -[UIView setNeedsLayout] at appropriate times
  - For instance, on size changes of your view

Demo

Laying Out Lights

UIView

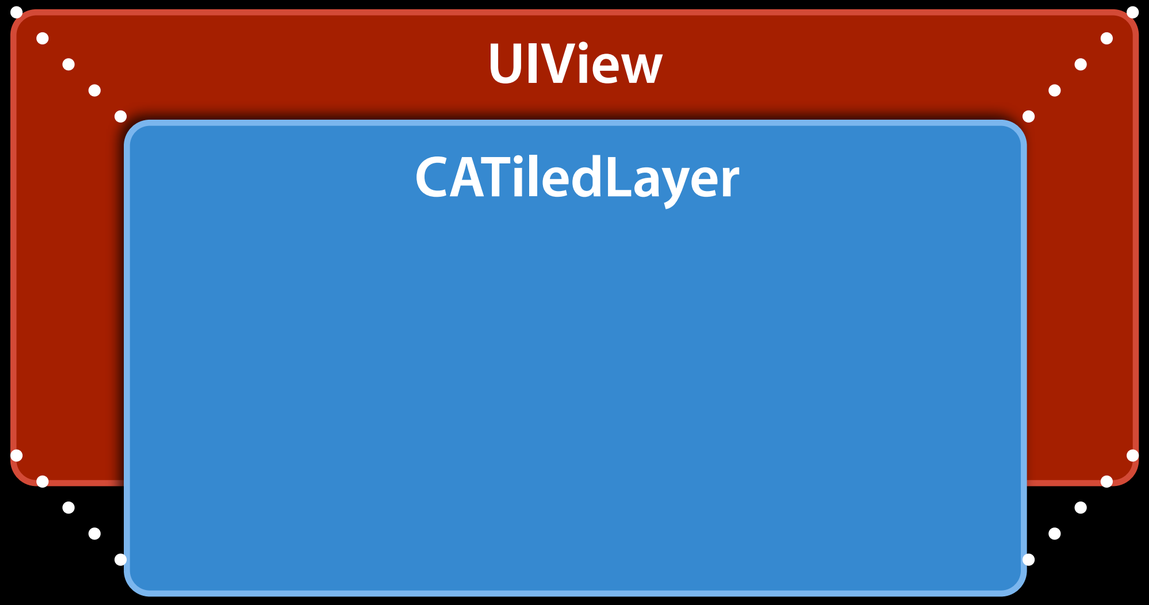
Getting the pretty on screen



- UIView is a thin wrapper around CALayer

UIView

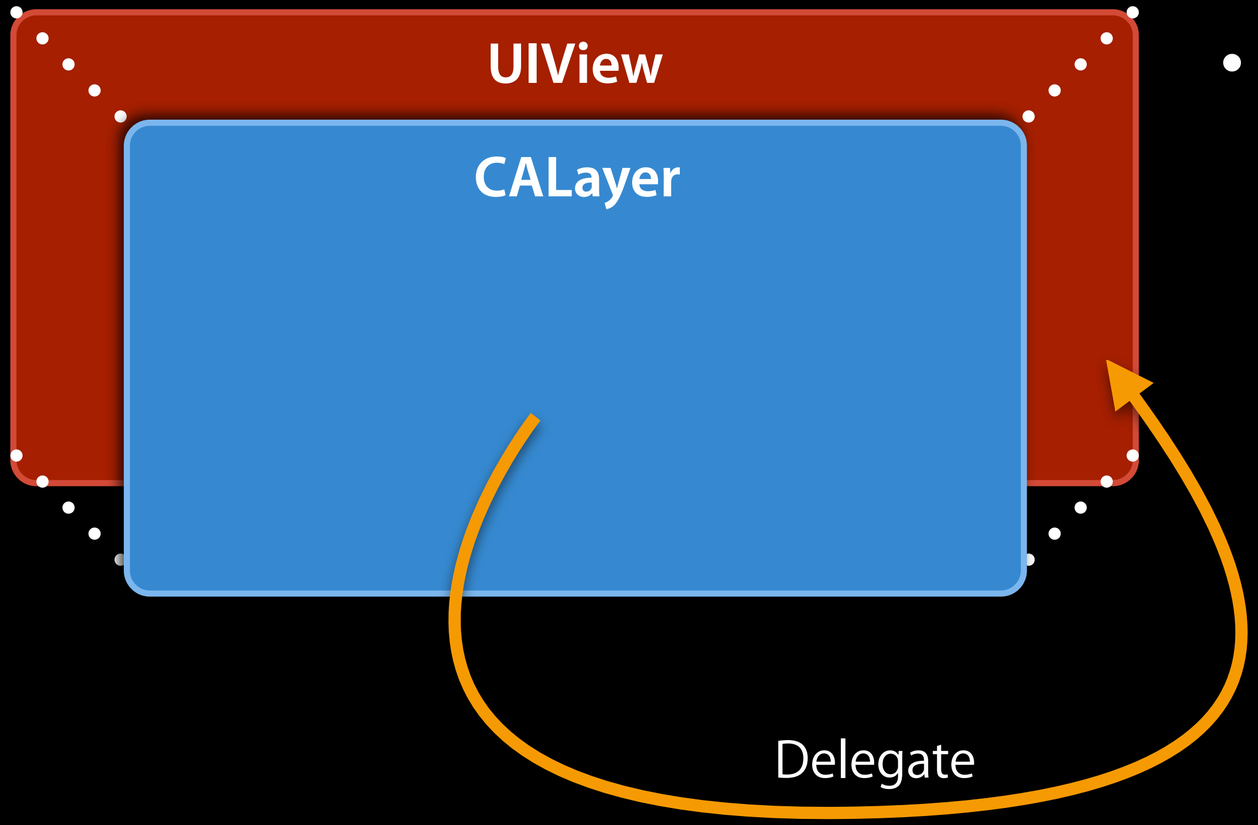
Getting the pretty on screen



- CALayer instance is readonly
  - CA class can be overridden by subclassing UIView and implementing
    +[UIView layerClass]

UIView

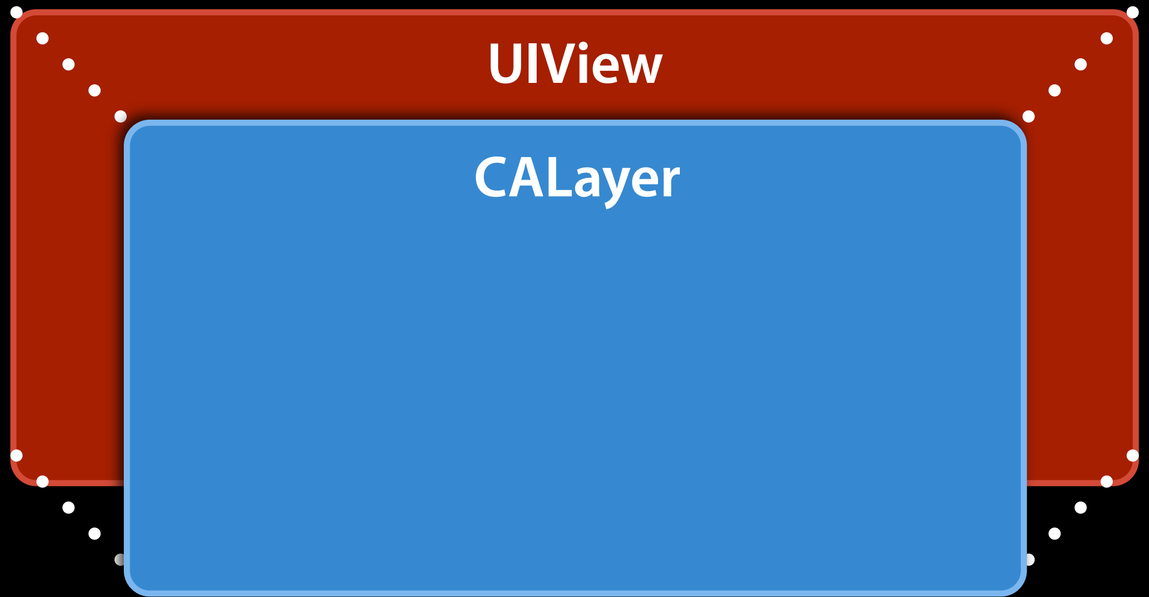
Getting the pretty on screen



- View is the layer's delegate
  - Don't set the layer's delegate!
  - Can subclass UIView and override delegate calls

UIView

Getting the pretty on screen



- One-to-one relationship
  - Frame
  - Bounds
  - Background color
  - etc…

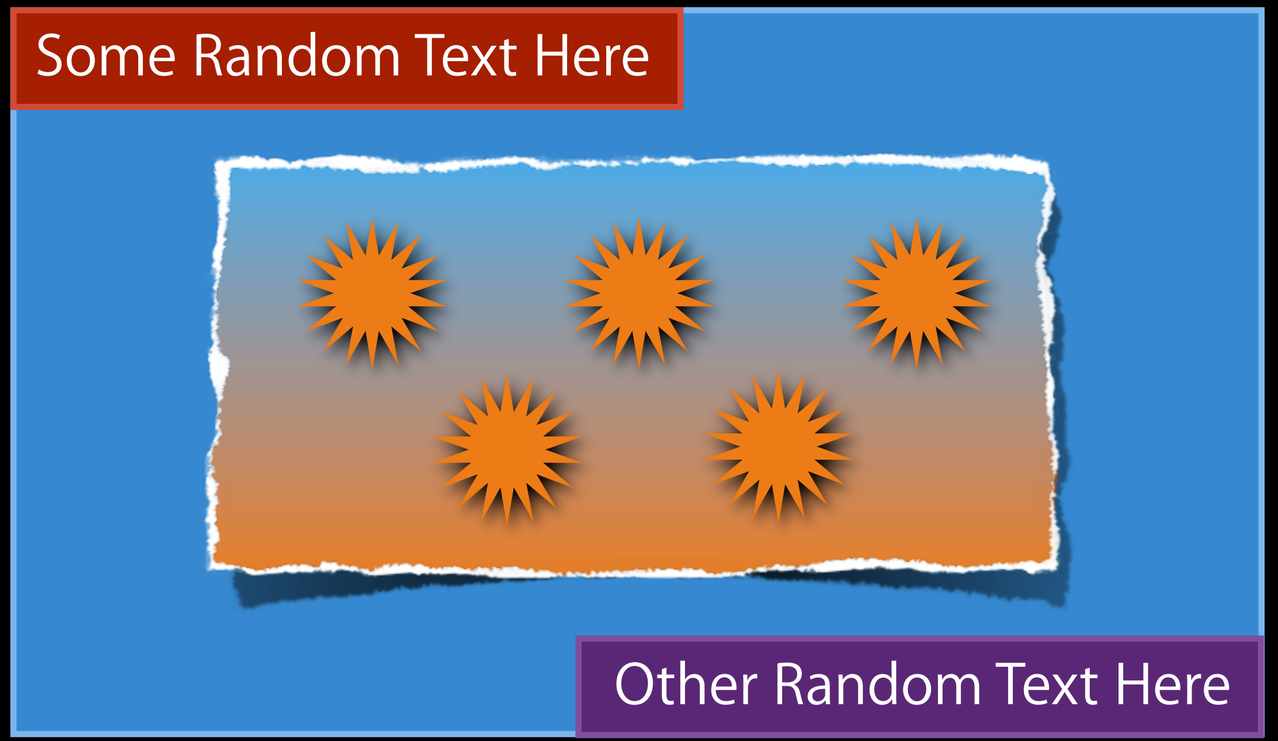
Drawing in a UIView

-[UIView drawRect:]

- Called by the system for you like -[UIView layoutSubviews]
  - Don’t ever call it yourself (except, again, to super as needed)
- Use -[UIView setNeedsDisplay] like -[UIView setNeedsLayout]
  - There is no corollary for -[UIView layoutIfNeeded]
- Should only be used for drawing within that view
  - Don’t do layout in drawRect
  - Don’t do expensive operations—draw and return

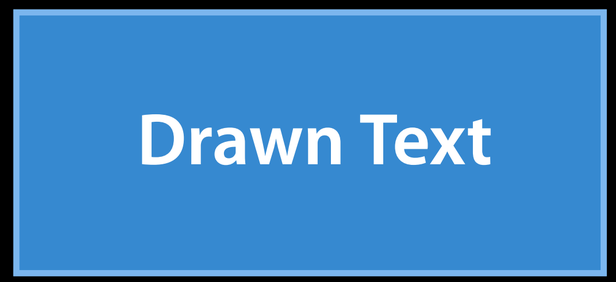
Drawing in a UIView

Complex drawing



Drawing in a UIView

Contents are cached “eternally”



Subtitle

-[UIView drawRect:]

-[UIView setClearsContextBeforeDrawing:NO]

Drawing in a UIView

Some hints / tricks

- UITableViewCell subclasses can not use -[UIView drawRect:]
  - Instead, add a custom subview to the -[UITableViewCell contentView] and draw there
- Use -[UIView setNeedsDisplayInRect:] to dirty particular regions
  - Not dirtying everything can save you significant time
- Use -[UIView drawRect:]'s rect parameter to your advantage
  - You only need to draw the area that has been dirtied

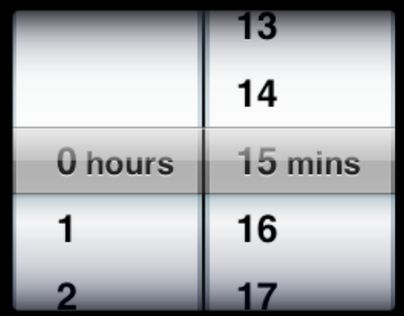
Perfecting Your iPhone Table Views	Pacific Heights Tuesday 2:00PM

Demo

Bringing the Lights

Controls

Use built-in controls



Built-In Controls

- Don’t subclass standard controls
  - Think about why you’re trying to do something different…
    …then redesign to be cohesive with other apps

Providing a Custom Look

- Use the -tintColor property
  - UIToolbar
  - UINavigationbar
  - UISearchBar
  - UISegmentedControl

- Use tintColor:nil to return to default

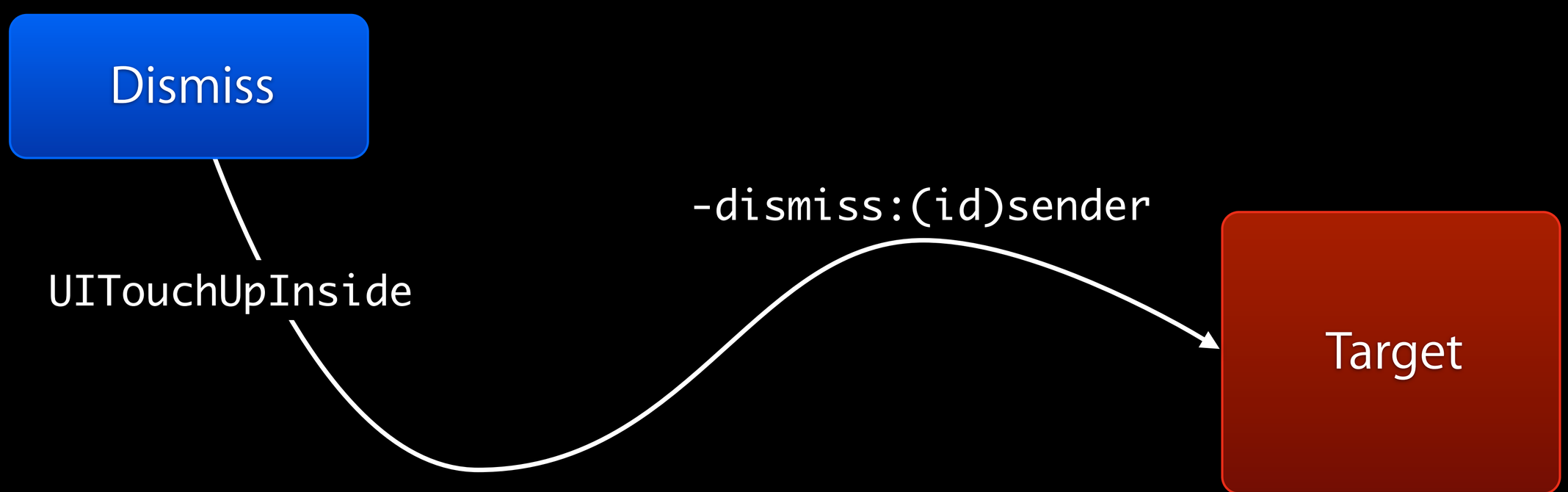
UIToolbar

- Use the provided toolbar items
  - The enum UIBarButtonSystemItem defines many
  - Use the provided graphics
    - Base your usage on description, not the current image—the image may change in a future release
  - Using built-in graphics lets you stay consistent with other apps

UIControl

- Targets respond to events

```
- (void)addTarget:(id)target action:(SEL)action forControlEvents:
  (UIControlEvents)controlEvents;
```

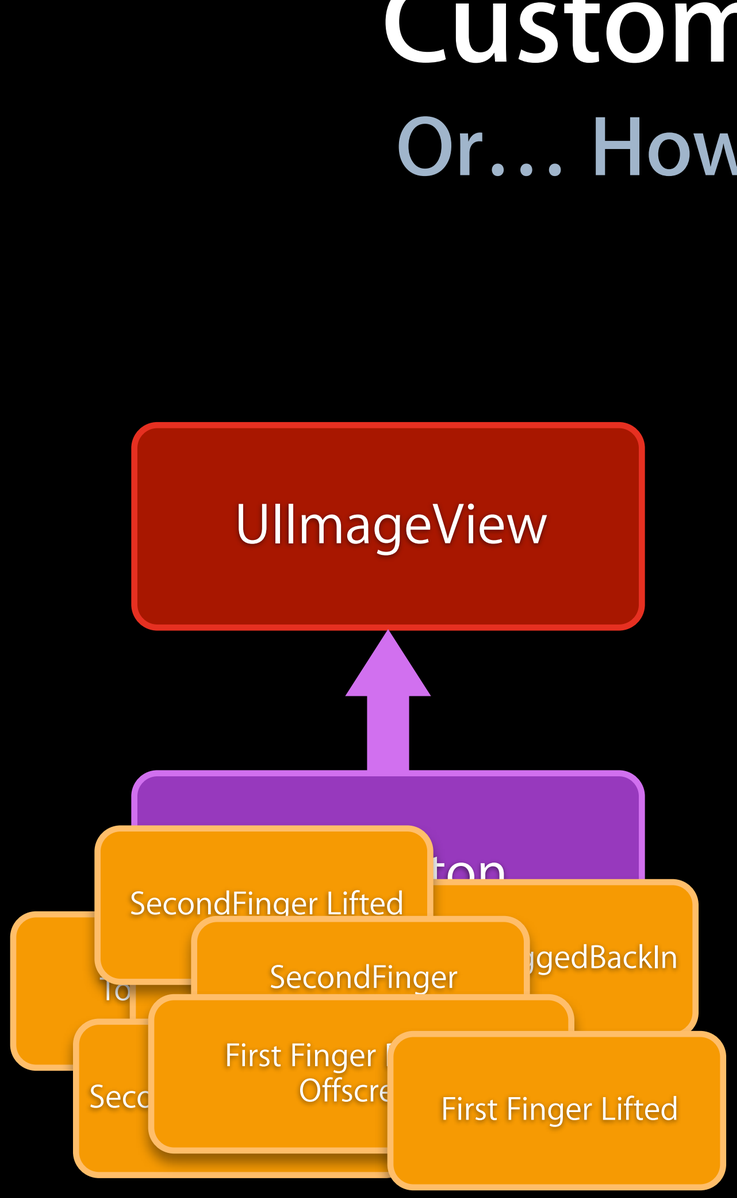


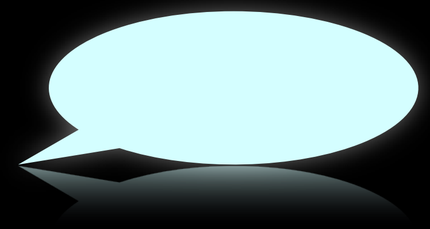
Custom Interactable Image

Or… How do I make my own button?

UIImageView

UIButton





-[UIButton setImage:forState:]

You've Fallen Asleep… It's Animation Time!

- `UIView(UIViewAnimation)` category
- No Animators—fast!
- Animations handled by Core Animation
- Core Animation uses HW acceleration

UIView Animations

Animation state is stacked

Set Frame

Begin Animation

Set Frame

Set Frame

Set Frame

Set Frame

Frame

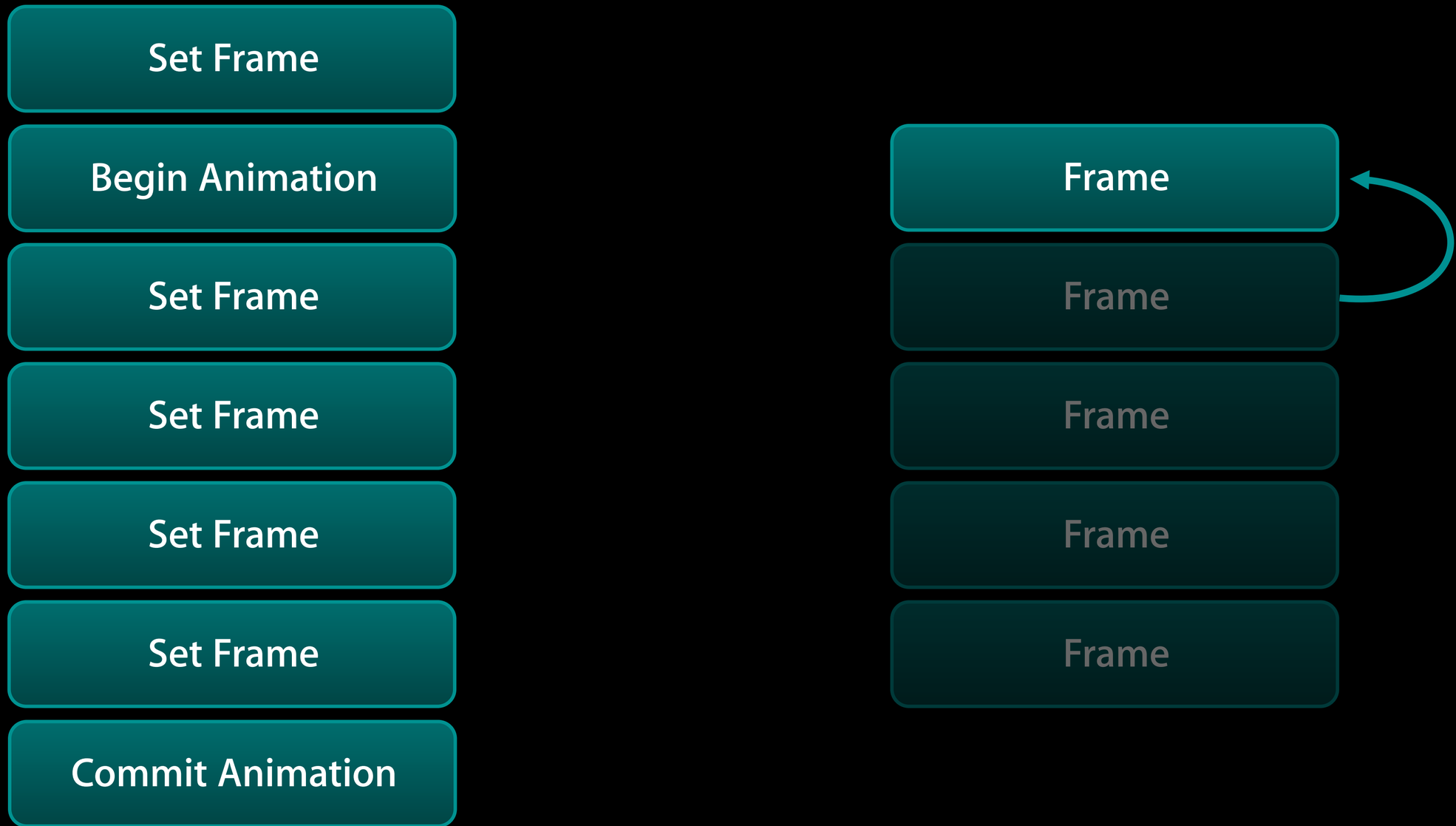
Frame

Frame

Frame

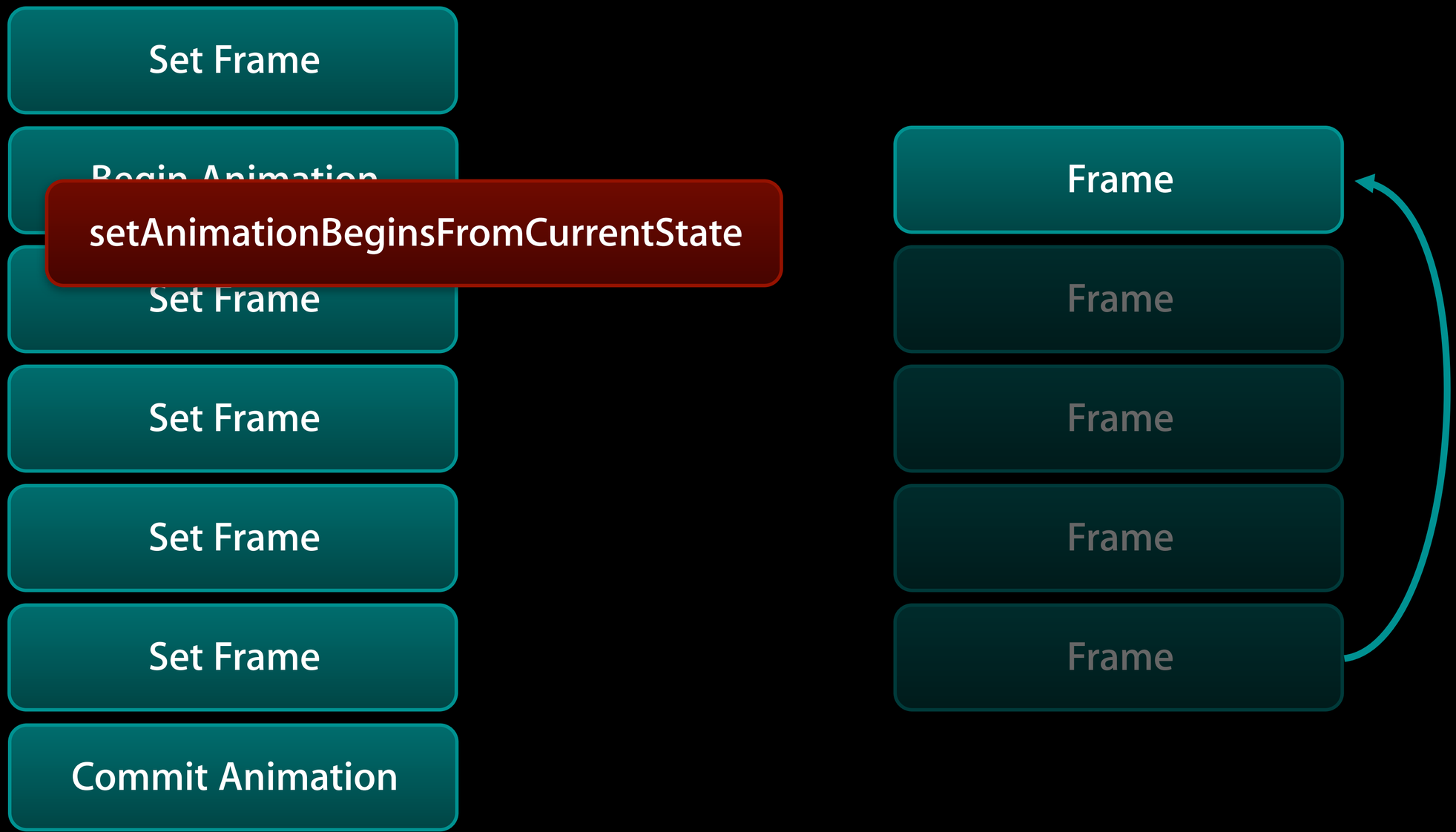
UIView Animations

Animation state is stacked



UIView Animations

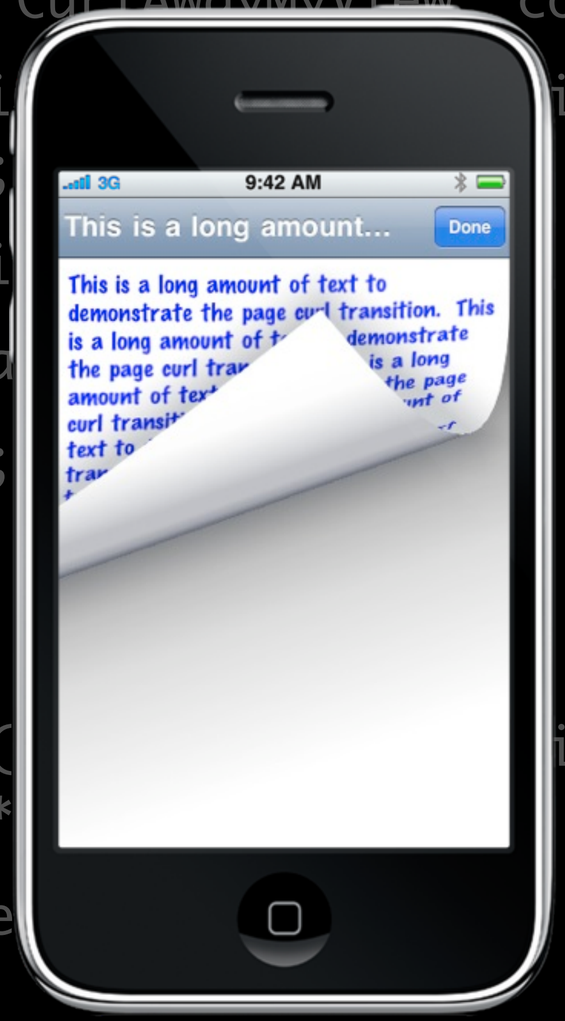
Animation state is stacked



UIView Transition Animations

```
[UIView beginAnimations:@"CurlAwayMyView" context:NULL];
[UIView setAnimationTransi                    ionTransitionCurlUp
forView:myView cache:YES];
[UIView setAnimationDurati
[UIView setAnimationDelega
[UIView commitAnimations];
[myView setAlpha:0.0f];


- (void)animationDidStop:(                    ionID finished:(NSNumber
*)finished context:(void *
    [myView removeFromSupe
}
```



Core Animation with UIViews

- Lower level animations possible with CA

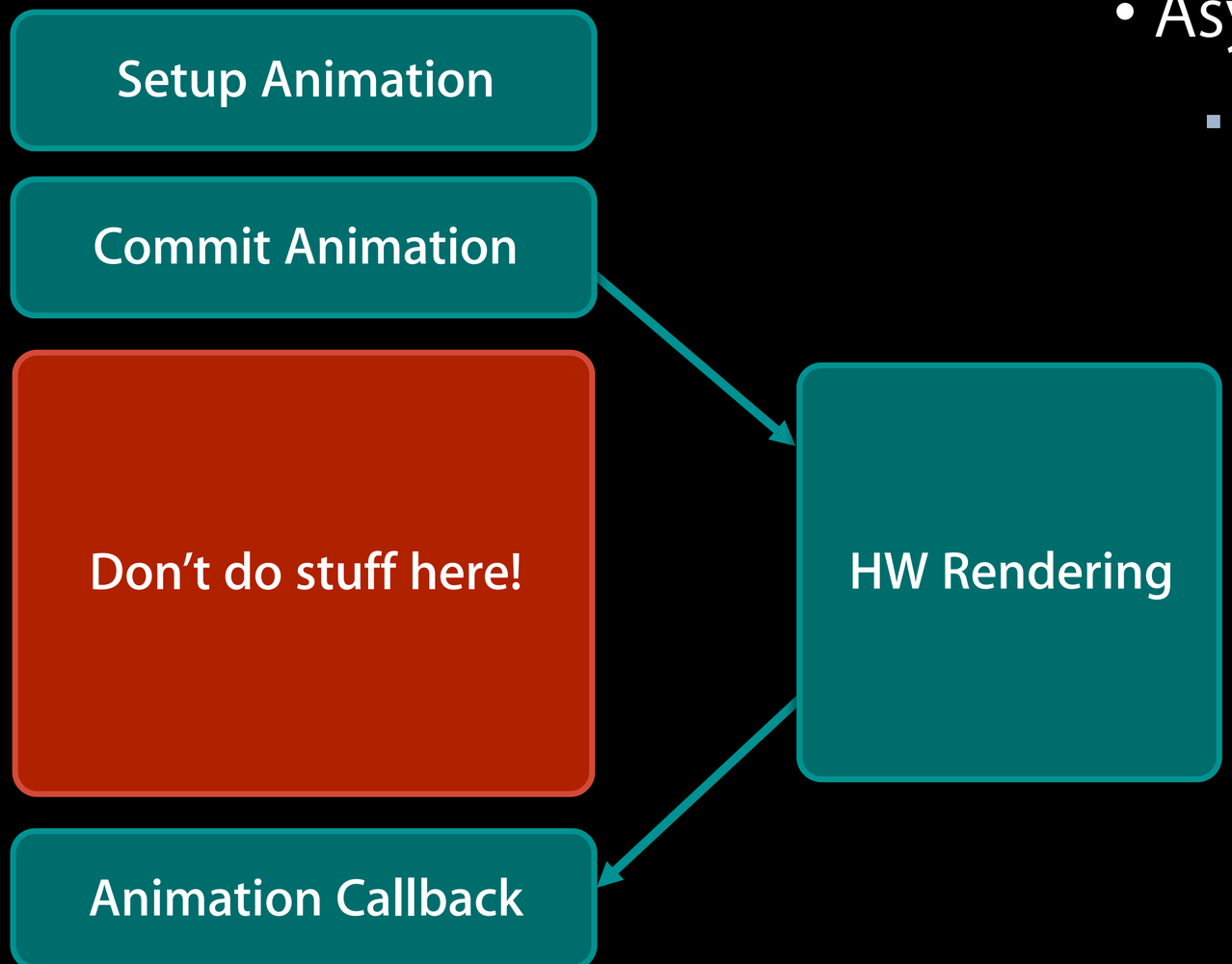
```
// CGPathRef path and UIView *view given

CAKeyframeAnimation *keyFrame = [CAKeyframeAnimation animation];
[keyFrame setKeyPath:@"position"];
[keyFrame setPath:path];
[keyFrame setDelegate:self];
[keyFrame setFillMode:@"frozen"];
[keyFrame setRemovedOnCompletion:NO];

[[view layer] addAnimation:keyFrame forKey:@"position"];
```

UIView Animations

- Asynchronous
  - Set up a delegate and callback



UIView Animations

- You don’t want the user interacting while things are moving on screen
  - What button are they touching? Do they know which button it is?
  - Is your app able to handle state changes while animating?
- Use -[UIApplication beginIgnoringInteractionEvents]
- Don’t forget to turn it off in your animation delegate

Demo

I Need New Paper, Stat!

Contact Slide

Matt Drance
Application Frameworks Evangelist
mdrance@apple.com

Documentation
iPhone Application Programming Guide
http://developer.apple.com/iphone

Related Sessions

Mastering iPhone Scroll Views	Pacific Heights Wednesday 2:00PM
Processing Multi-Touch Events on iPhone	Pacific Heights Thursday 9:00AM
Optimizing Performance on iPhone	Pacific Heights Thursday 10:30AM
Effective iPhone App Architecture	Presidio Thursday 3:30PM
iPhone View Controller Techniques	Presidio Thursday 5:00PM

Q&A

WWDC Session Survey

You can rate your sessions on the WWDC Attendee Site.

Survey forms are located on each session description page at:

developer.apple.com/wwdc/attendee

WWDC09